



Capability Maturity Model Overview

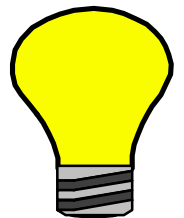
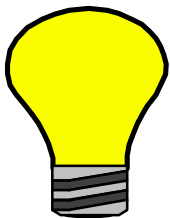
Presented by: DSDC

For more info, send requests to: sepg@dsdc.dla.mil

Course Objectives

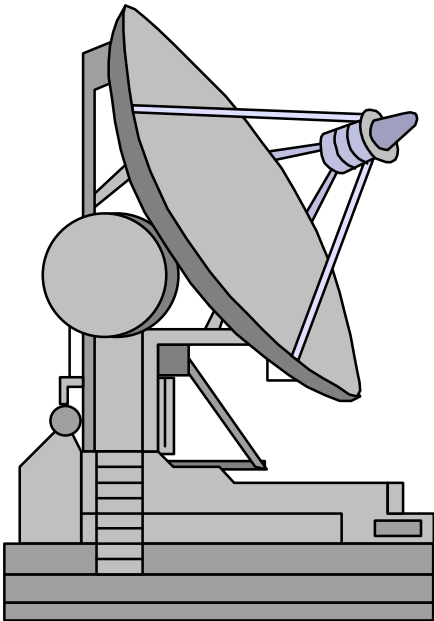
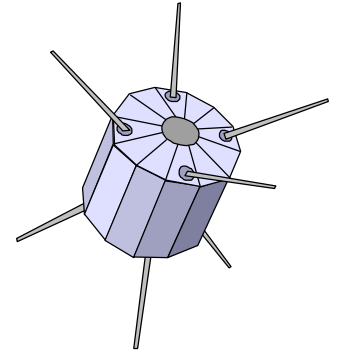
This course will introduce the student to the Software Process Capability Maturity Model. At the end of the course, the student will be able to:

- **Identify the terms used with the Capability Maturity Model.**
- **Recognize the elements of the Capability Maturity Model.**
- **Recognize each of the five levels of the Capability Maturity Model**
- **Recognize how the CMM fits in with current and future organizational information systems development projects.**





DoD Objectives for its software community to achieve by the year 2000

- ① Reduce life-cycle costs by a factor of two**
- ② Reduce software problem rates by a factor of ten**
- ③ Achieve new levels of DoD mission capability and interoperability via software**



Additional Readings

 *Capability Maturity Model for Software*, Software Engineering Institute,
CMU/SEI-93-TR-24, Aug '91, Feb '93

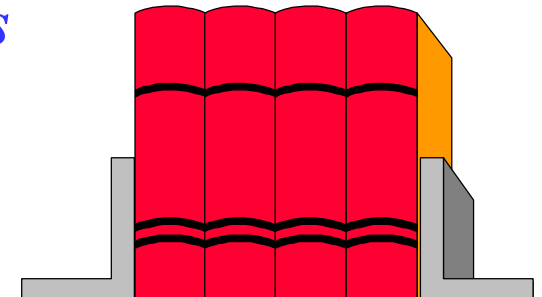
 *Key Practices of the Capability Maturity Model*, Software Engineering Institute,
CMU/SEI-93-TR-25, Aug '91, Feb '93

BOTH AVAILABLE AT:

http://www.dataweb.net/~lodewijk/SEI_CMM/

 *Managing the Software Process*, Watts S. Humphrey,
Addison Wesley, 1990

 *The Capability Maturity Model: Guidelines for Improving the Software Process*, Mark Paulk et al, Addison-Wesley, 1995





Software Process Management

Define Software:

Define Process:

Define Software Process Improvement:

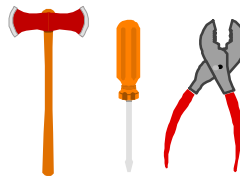
Software Process Management

Software Engineering Institute's Definition of Process

The means by which people, procedures, methods, equipment, and tools are integrated to produce a desired end result.



People with skills,
training, and motivation



Tools and
equipment



Procedures and methods
defining the relationships
of tasks

Software Process Management

Life Cycle Phases:

⇒ **Initiation**

⇒ **Development**

- Analysis
- Design
- Programming

⇒ **Evaluation**

⇒ **Operations**





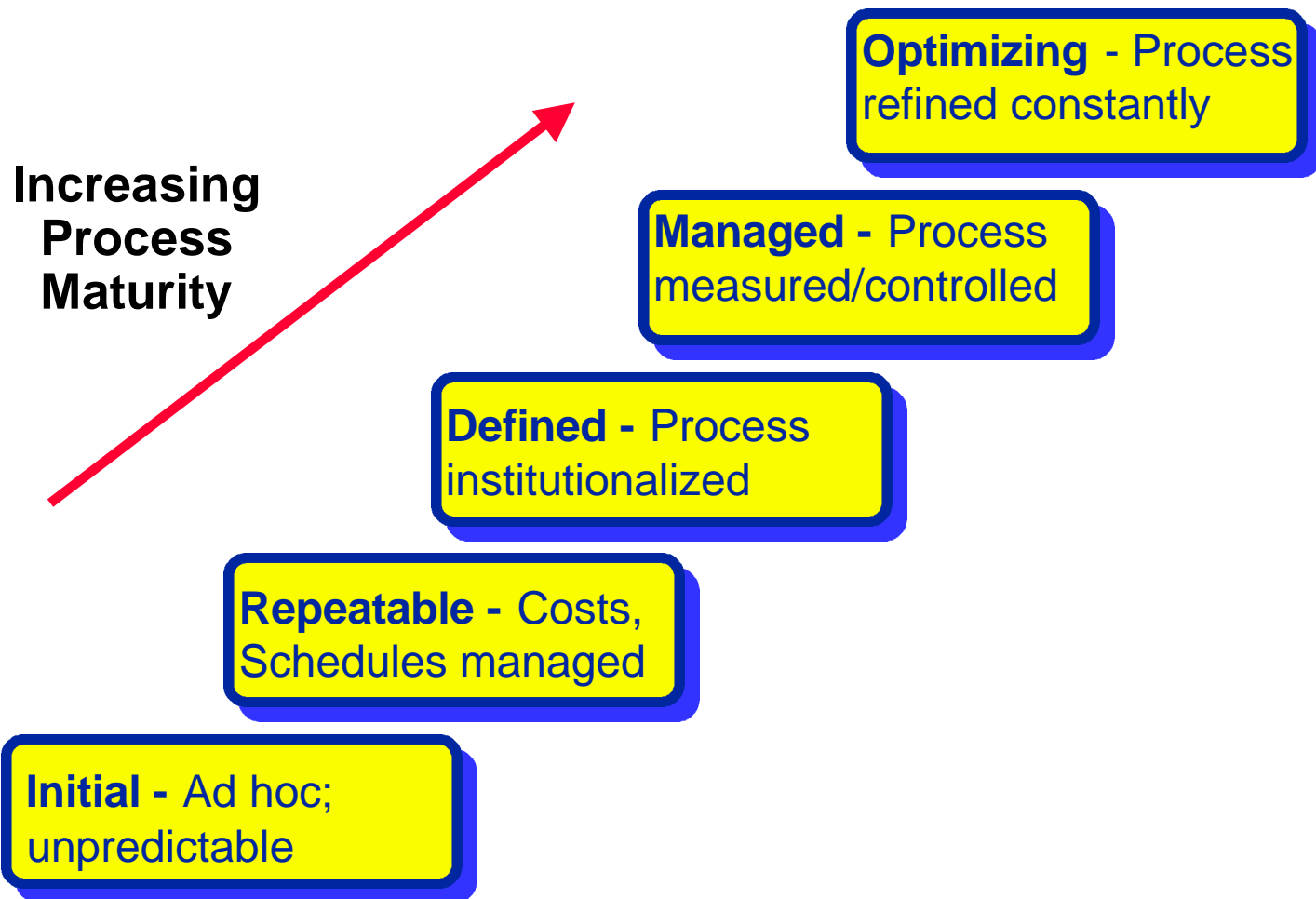
Software Process Management

To Achieve Improvement

The Capability Maturity Model for Software (CMM) is a five-level roadmap for improving the software process and achieving improved quality results.

Software Process Management

Capability Maturity Model Overview





The Capability Maturity Model

What is the Capability Maturity Model (CMM)?

The application of process management and quality improvement concepts to software development and maintenance.

A guide for evolving toward a culture of engineering excellence.

A model for organizational improvement.



The Capability Maturity Model

The CMM Community

CMM provides a principled, public model for appraising software development capability.

CMM owned by the international software community.

SEI exercises stewardship over the CMM

- **evolves CMM from industry and government experience**
- **performs empirical research**
- **develops CMM-based improvement methods**



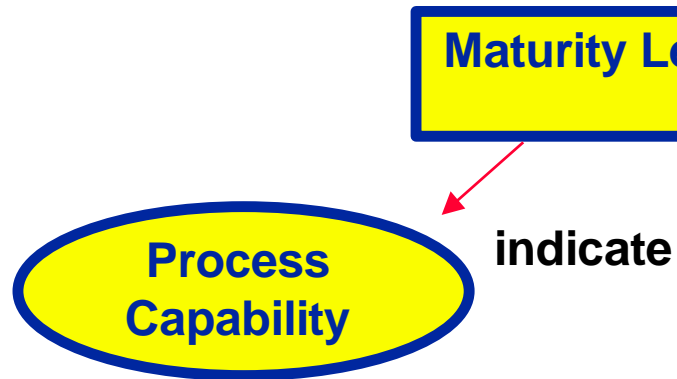
The Capability Maturity Model

The CMM Structure

Maturity Levels

The Capability Maturity Model

The CMM Structure



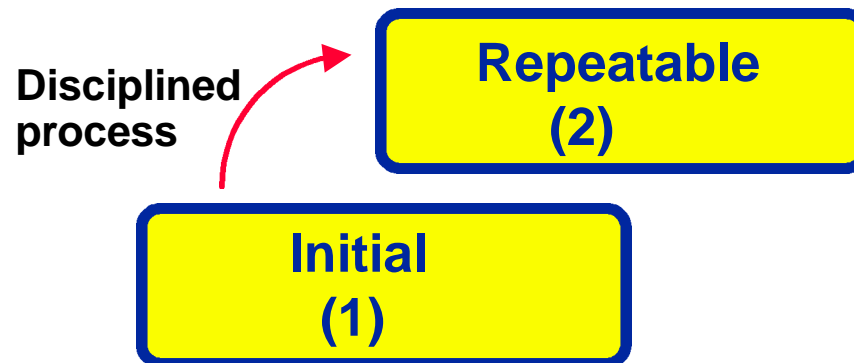
The Capability Maturity Model

The Five Maturity Levels

**Initial
(1)**

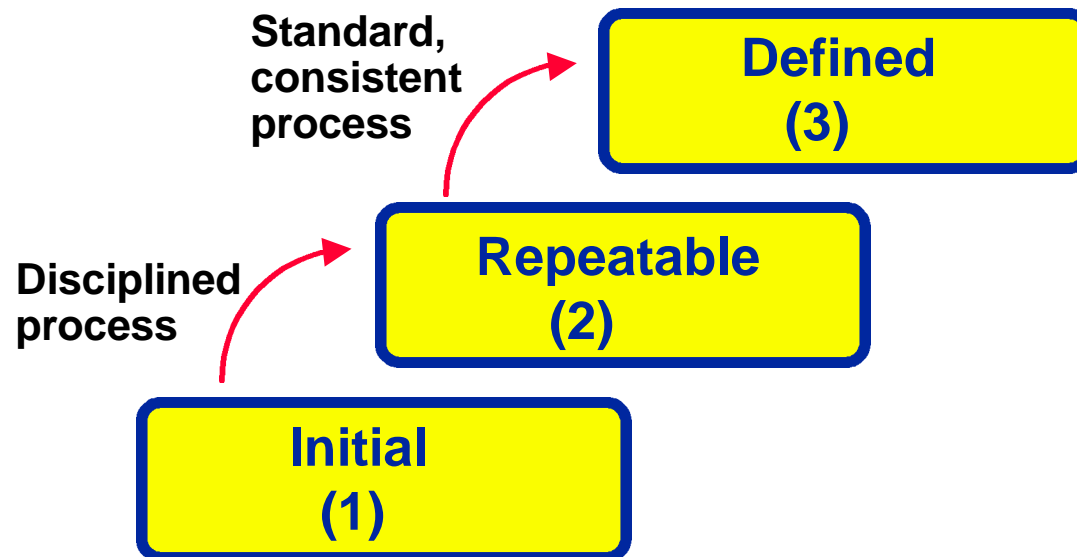
The Capability Maturity Model

The Five Maturity Levels



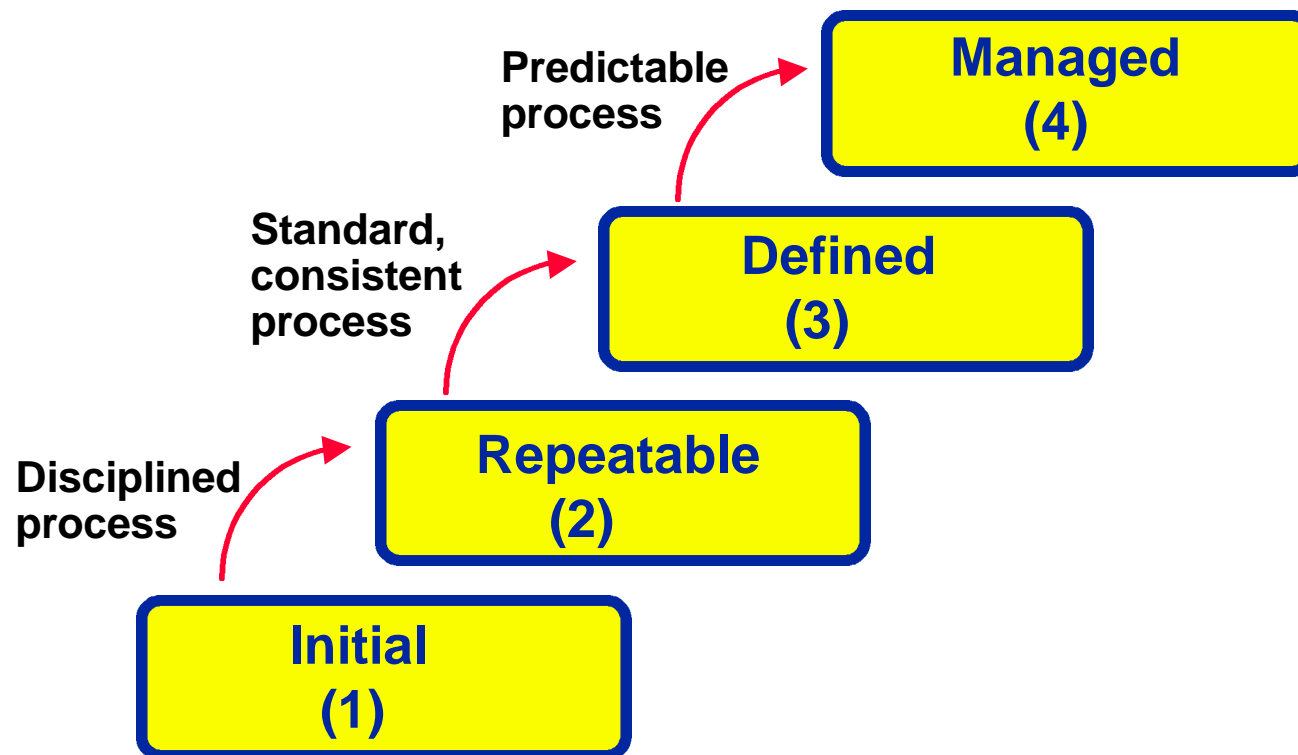
The Capability Maturity Model

The Five Maturity Levels



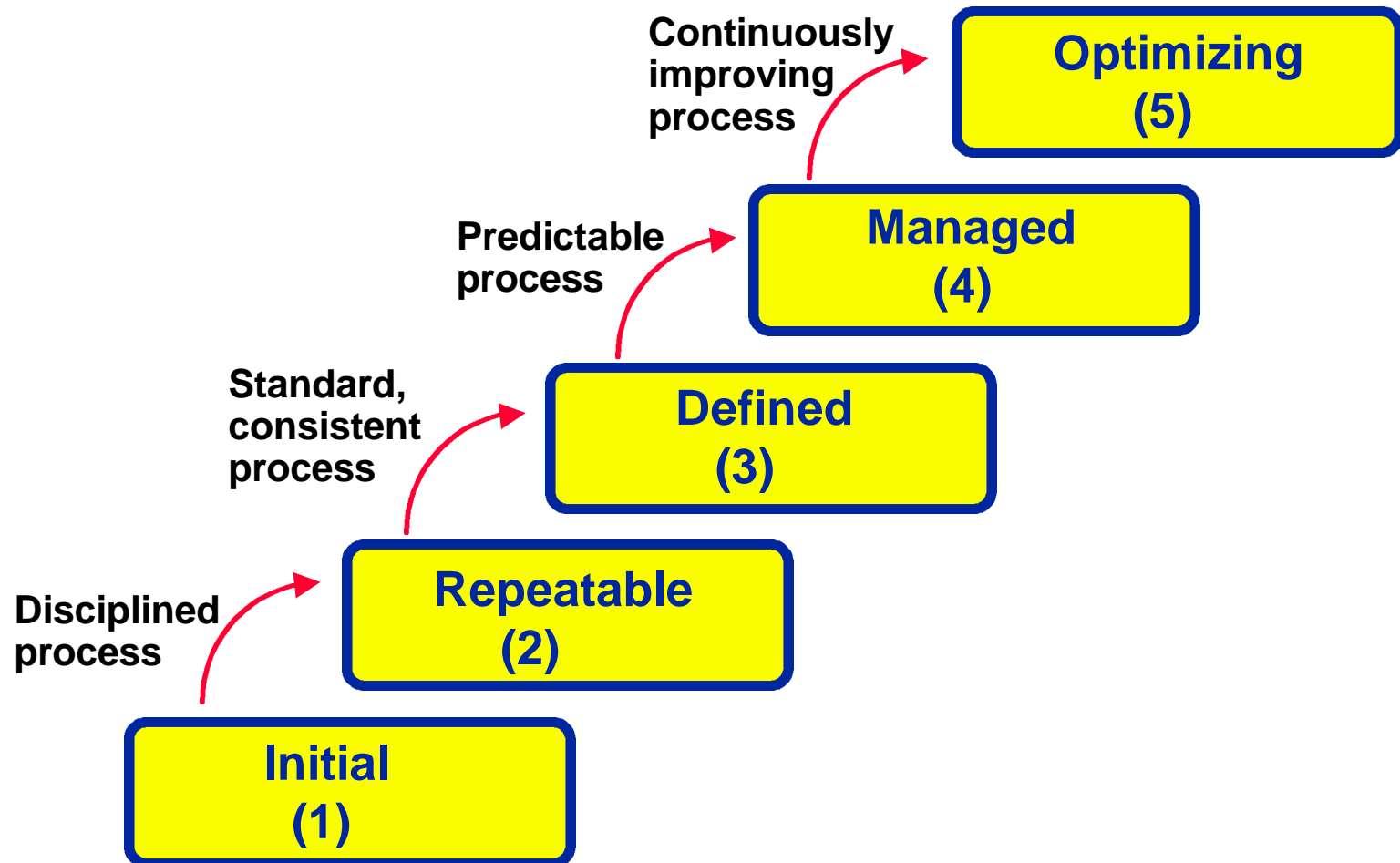
The Capability Maturity Model

The Five Maturity Levels



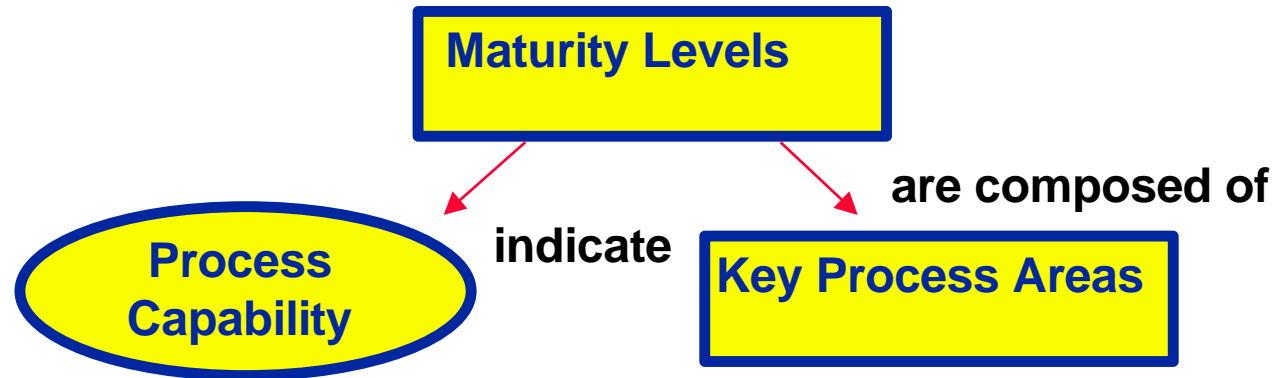
The Capability Maturity Model

The Five Maturity Levels



The Capability Maturity Model

The CMM Structure



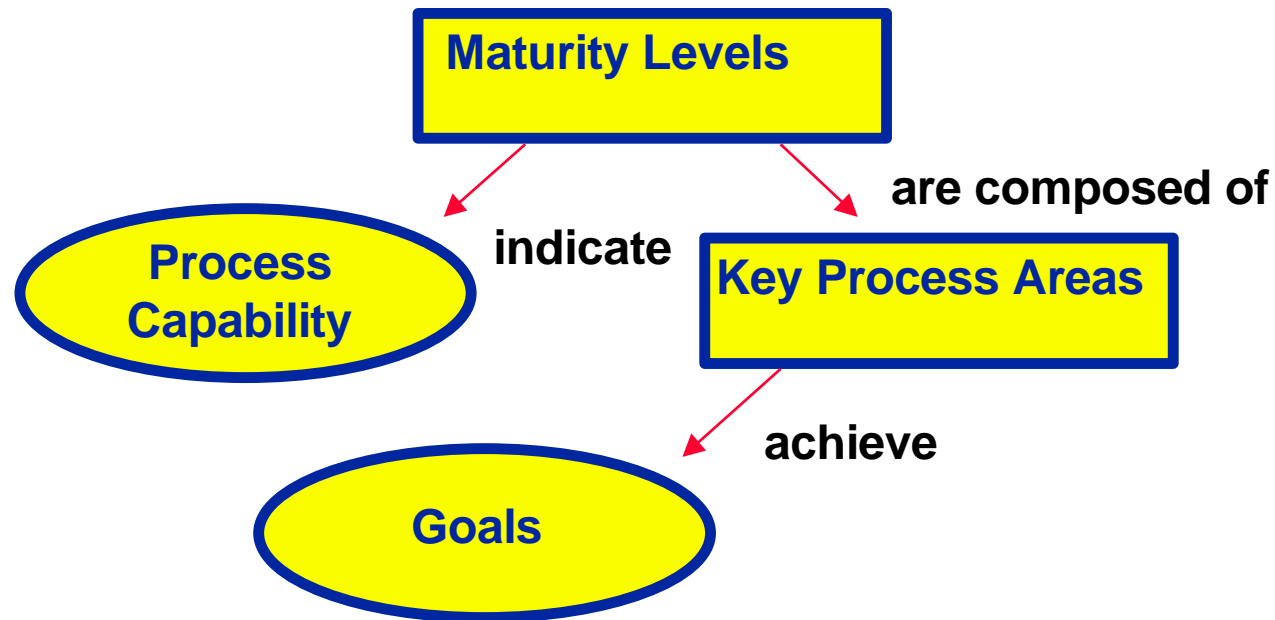
The Capability Maturity Model

Focus of the Key Process Areas

Level	Focus	Key Process Area
Optimizing (5)	Continuous Process improvement	Defect Prevention Technology Change Management Process Change Management
Managed (4)	Product & process quality	Quantitative process management Software quality management
Defined (3)	Engineering process	Organization process focus Organization process definition Training program Integrated software management Software product engineering Intergroup coordination Peer Reviews
Repeatable (2)	Project management	Requirements management Software project planning Software project tracking & oversight Software subcontract management Software quality assurance Software configuration management
Initial (1)		

The Capability Maturity Model

The CMM Structure



The Capability Maturity Model

An Example of Goals: Software Project Planning

Software estimates are documented for use in planning and tracking the software project.

Software project activities and commitments are planned and documented.

Affected groups and individuals agree to their commitments related to the software project.

The Capability Maturity Model

Example Maturity Questions on Size Estimating

Do you use a documented procedure to estimate software size (e.g. lines of code, function points, etc..)?

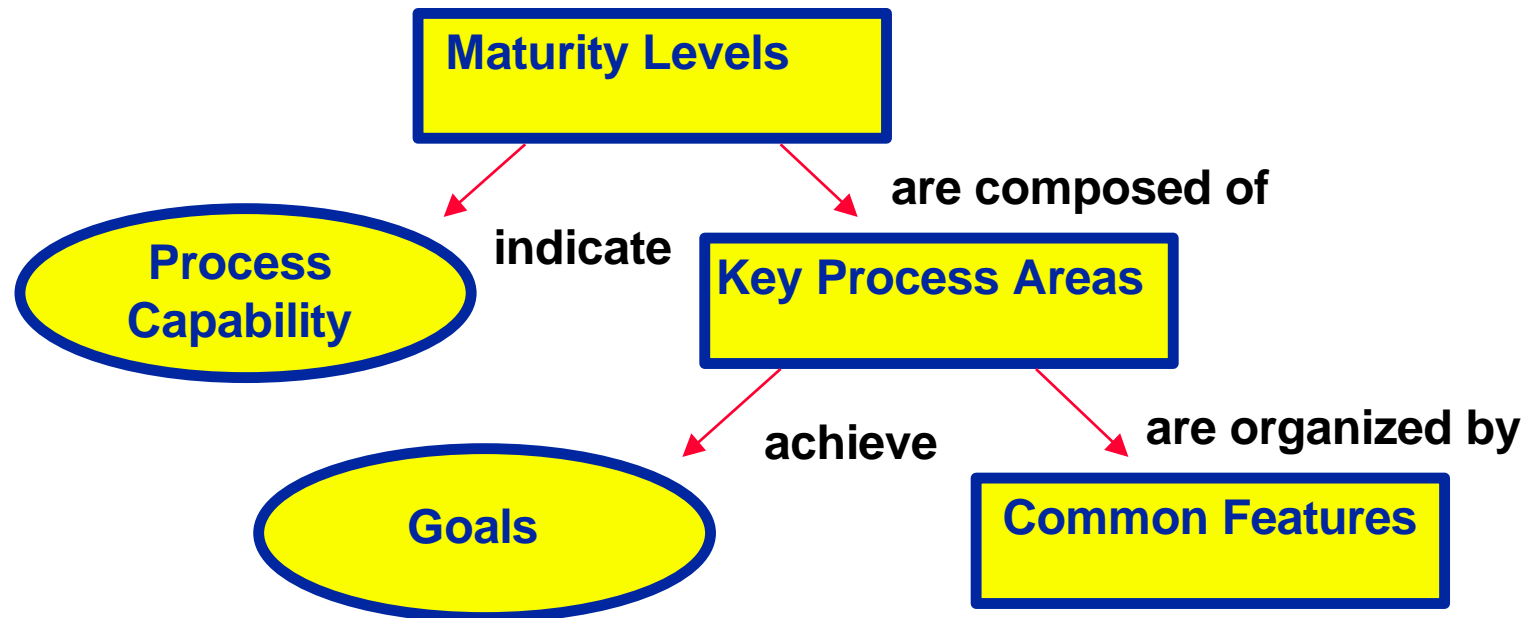
Do you use historical size data, when available, to help derive software size estimates?

Do you document the assumptions made in estimating software size?

Do you review software size estimates?

The Capability Maturity Model

The CMM Structure



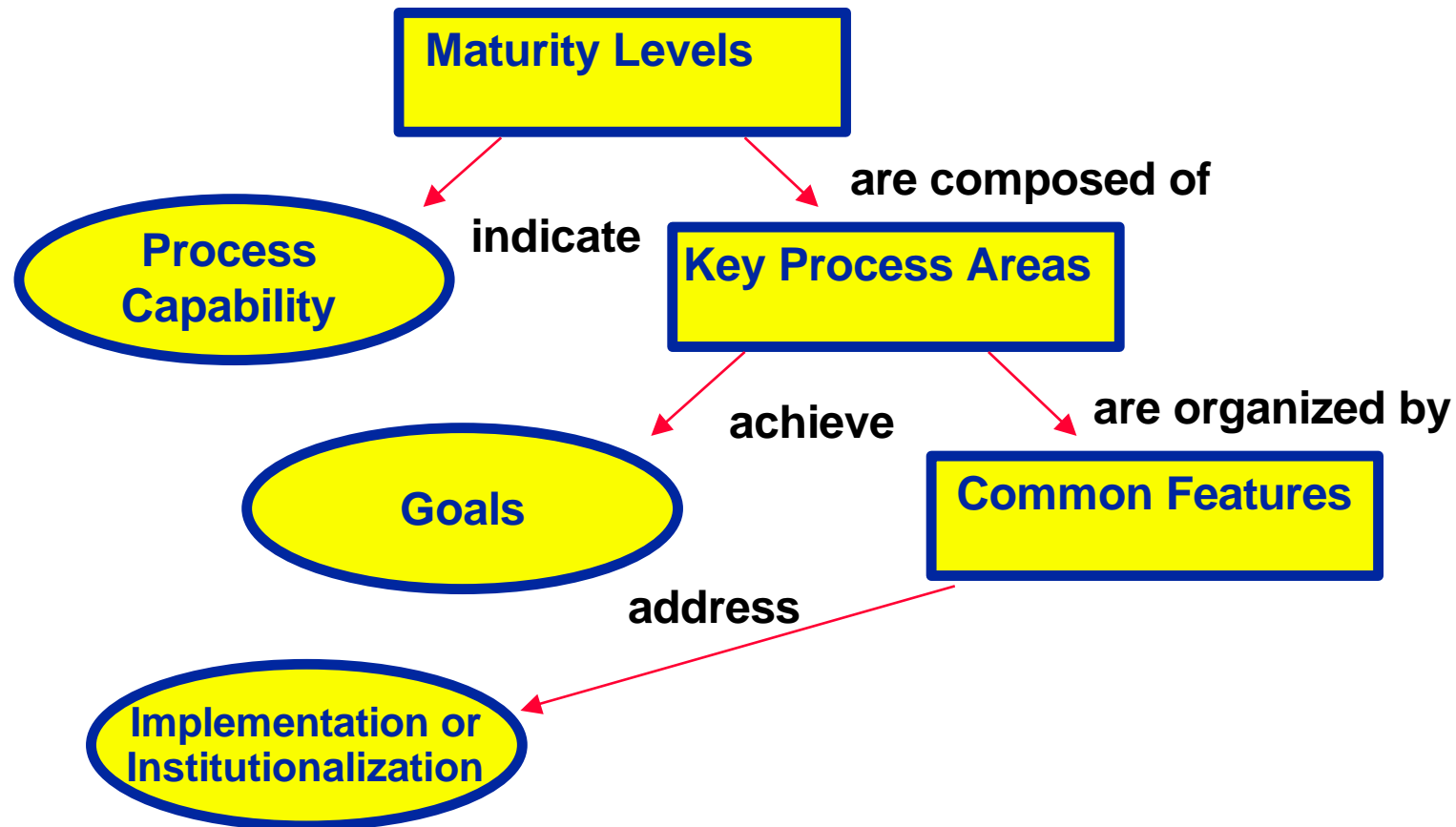
The Capability Maturity Model

Key Process Area Common Features

- **Commitment to Perform**
- **Ability to Perform**
- **Activities Performed**
- **Measurement and Analysis**
- **Verifying Implementation**

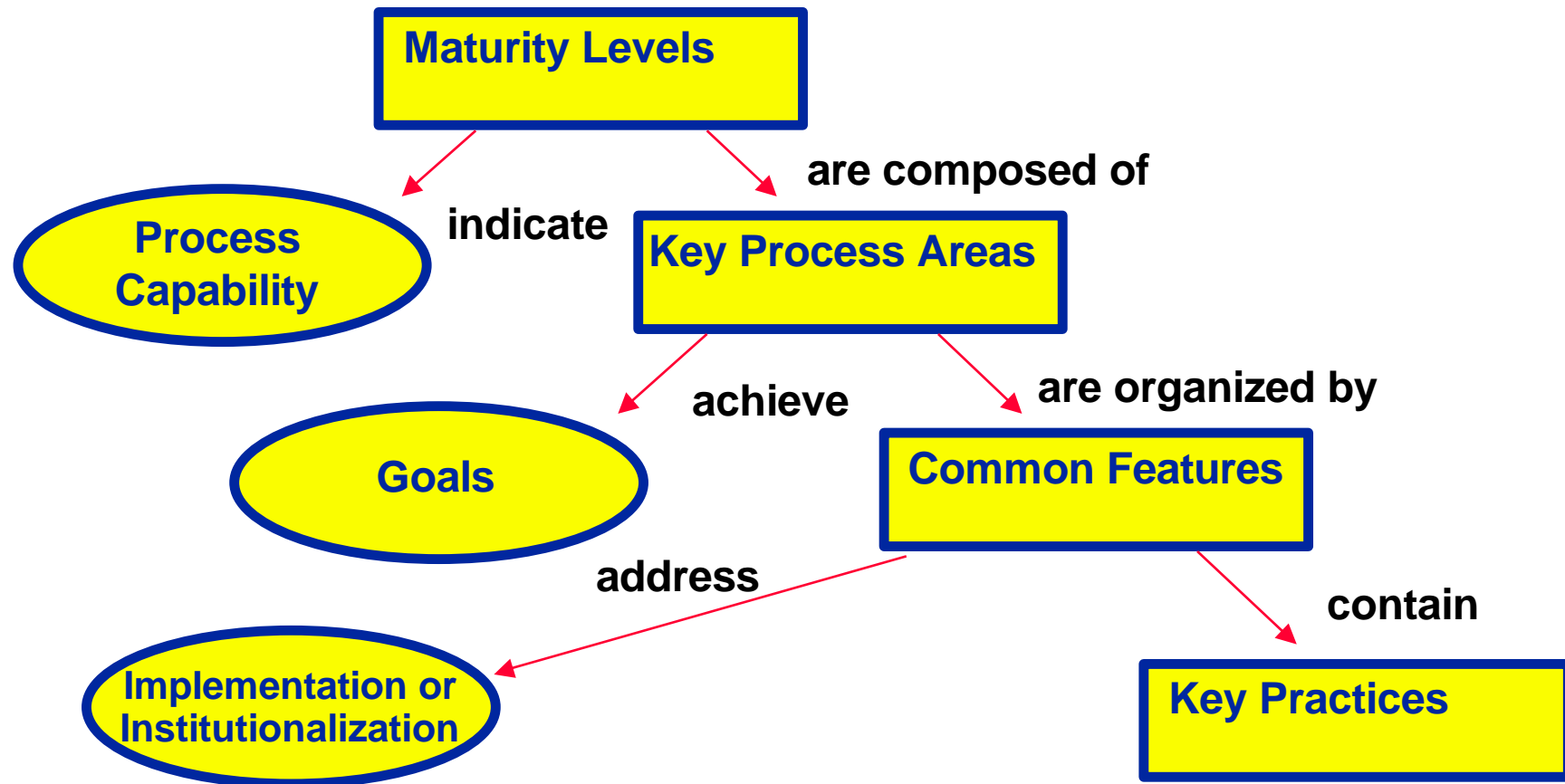
The Capability Maturity Model

The CMM Structure



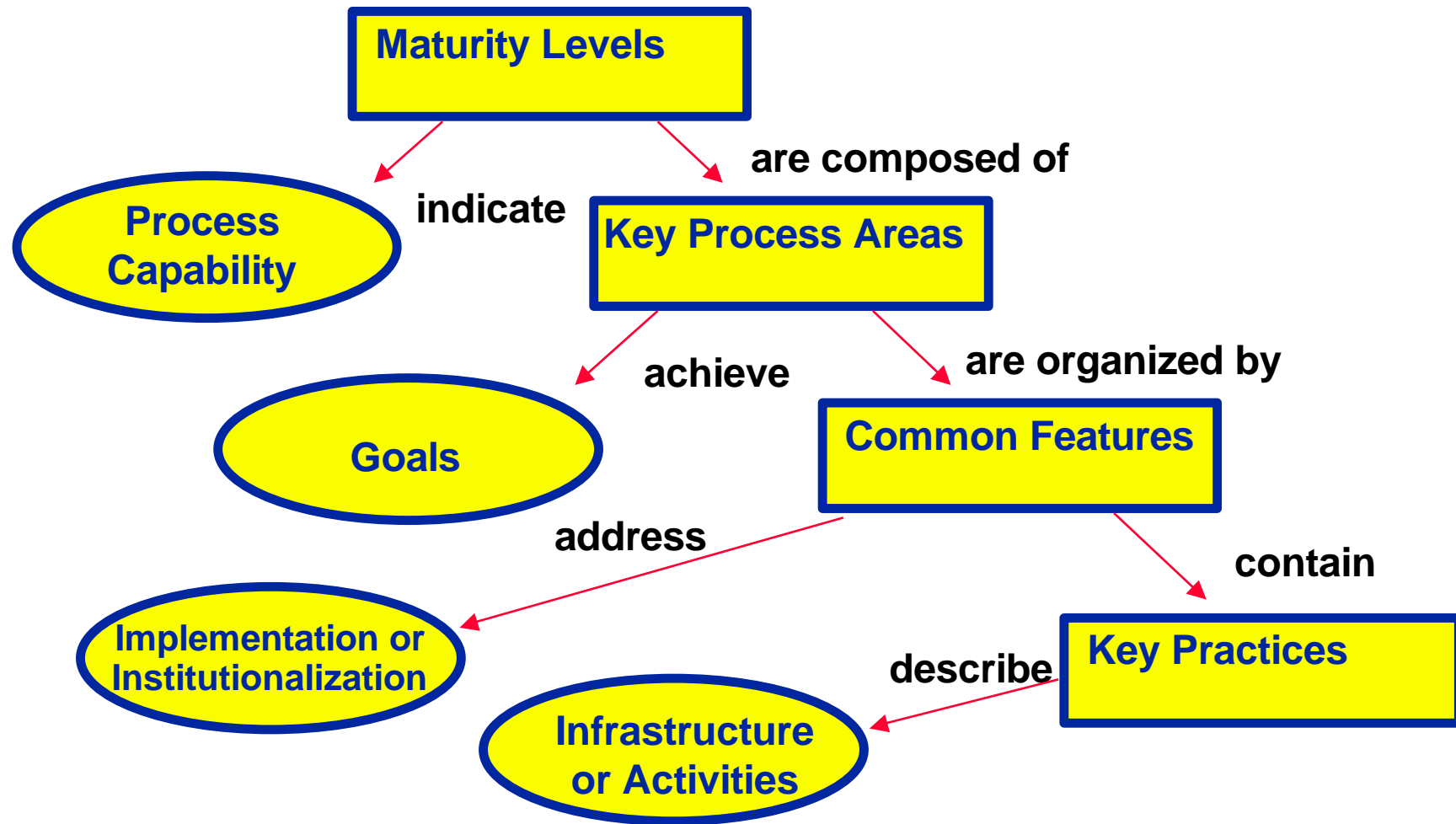
The Capability Maturity Model

The CMM Structure



The Capability Maturity Model

The CMM Structure



The Capability Maturity Model

An Example Key Practice: Size Estimating

Software Project Planning

Activity 9 Estimates for the size of the software work products (or changes to the size of software work products) are derived according to a documented procedure:

This procedure typically specifies that



The Initial Maturity Level

Understanding the Initial Maturity Level

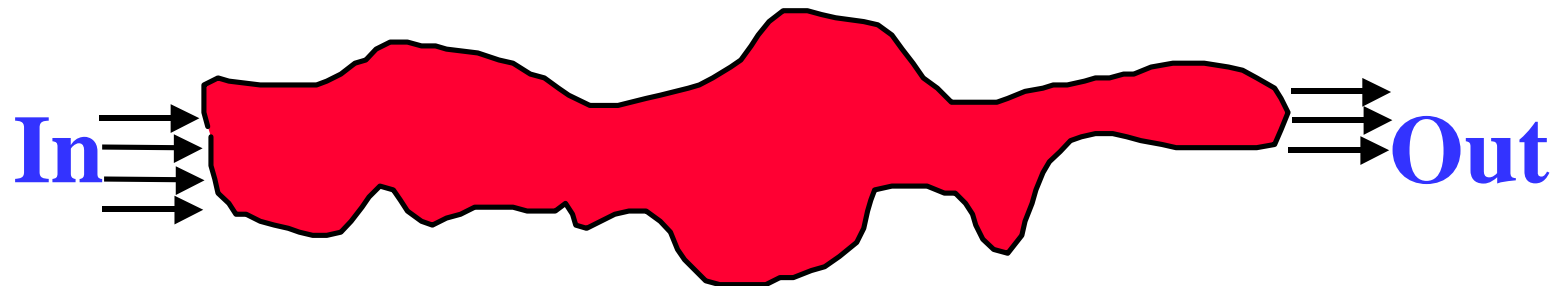
Performance driven by the competence and heroics of the people doing the work.

Consistency and compliance to standards driven by management priorities - usually schedule is the top priority.

High quality and exceptional performance possible so long as the best people can be hired.

The Initial Maturity Level

The Management View of the Software Process at Level 1



Requirements flow in.

A software product is (usually) produced by some amorphous process.

The product flows out and (hopefully) works.

The Repeatable Maturity Level

Understanding the Repeatable Maturity Level

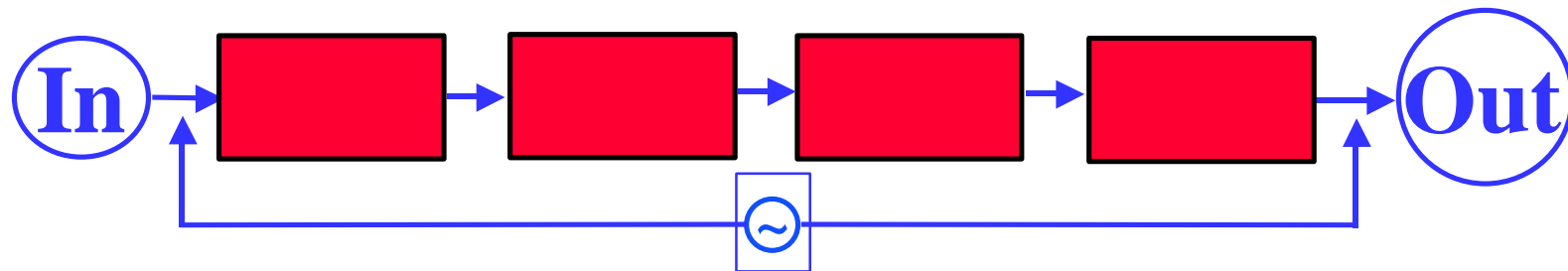
The major problems in software development are managerial - not technical.

Management must “walk their talk” to initiate an improvement effort.

Without management discipline, good software engineering practices typically are abandoned in the crunch.

The Repeatable Maturity Level

The Management View of the Software Process at Level 2



Requirements and resources flow in.

The production of the software product is visible at defined points.

Artifacts of the process are controlled.

The Repeatable Maturity Level

The Key Process Areas for the Repeatable Level (2)

Repeatable (2)

Software configuration management
Software quality assurance
Software subcontract management
Software Project Tracking and Oversight
Software Project Planning
Requirements Management

The Defined Maturity Level

Understanding the Defined Maturity Level

To control a process, it must be well understood.

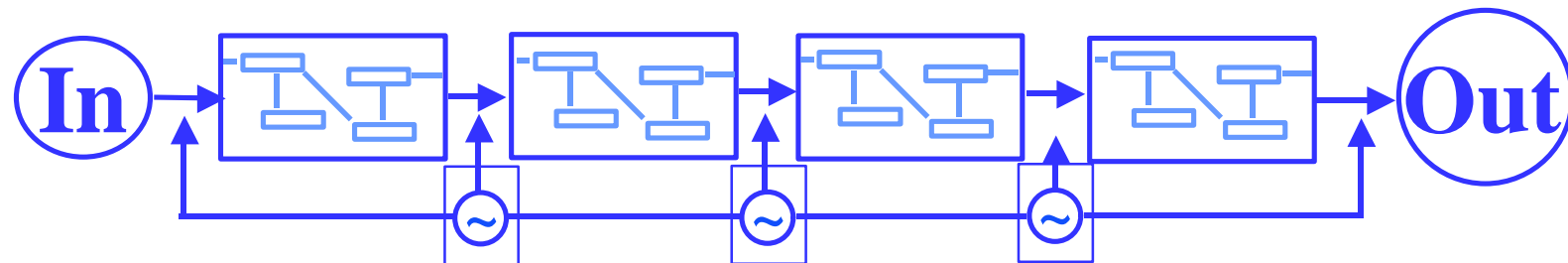
Identify the inputs, how they will affect the process, and their readiness criteria.

Identify the outputs and the completion criteria for the outputs.

Establish a shared understanding of how the process works and the role of each participant.

The Defined Maturity Level

The Management View of the Software Process at Level 3



Roles and responsibilities in the process are understood.

The production of the software product is visible throughout the software process.

The Defined Maturity Level

The Key Process Areas for the Defined Level (3)

Defined (3)

Peer Reviews
Intergroup coordination
Software product engineering
Integrated software management
Training program
Organization process definition
Organization process focus



The Managed Maturity Level

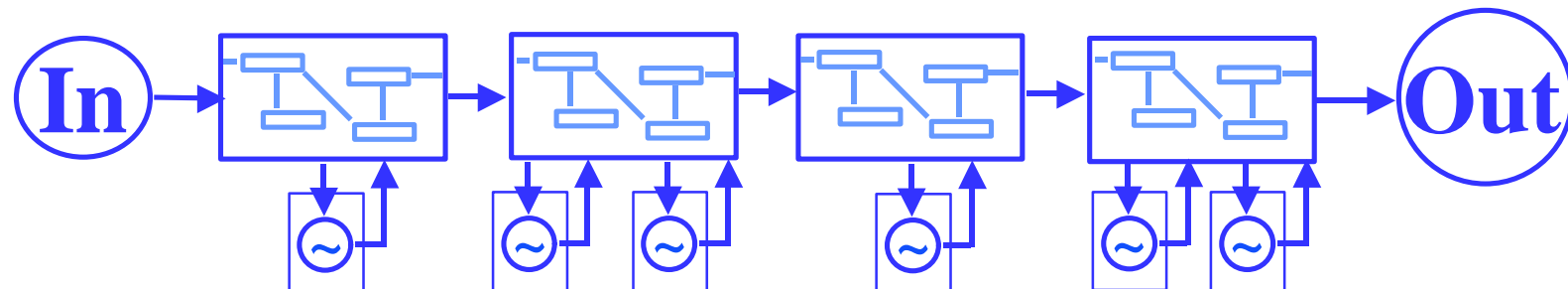
Understanding the Managed Maturity Level

Applying the principles of statistical process control, address special causes of process variation.

Quantitatively address the organization's, customer's and end user's quality goals as part of a philosophy of quality management.

The Managed Maturity Level

The Management View of the Software Process at Level 4



The production of the software product is quantitatively understood throughout the software process.

The Managed Maturity Level

The Key Process Areas for the Managed Level (4)

Managed (4)

**Software quality management
Quantitative process management**



The Optimizing Maturity Level

Understanding the Optimizing Maturity Level

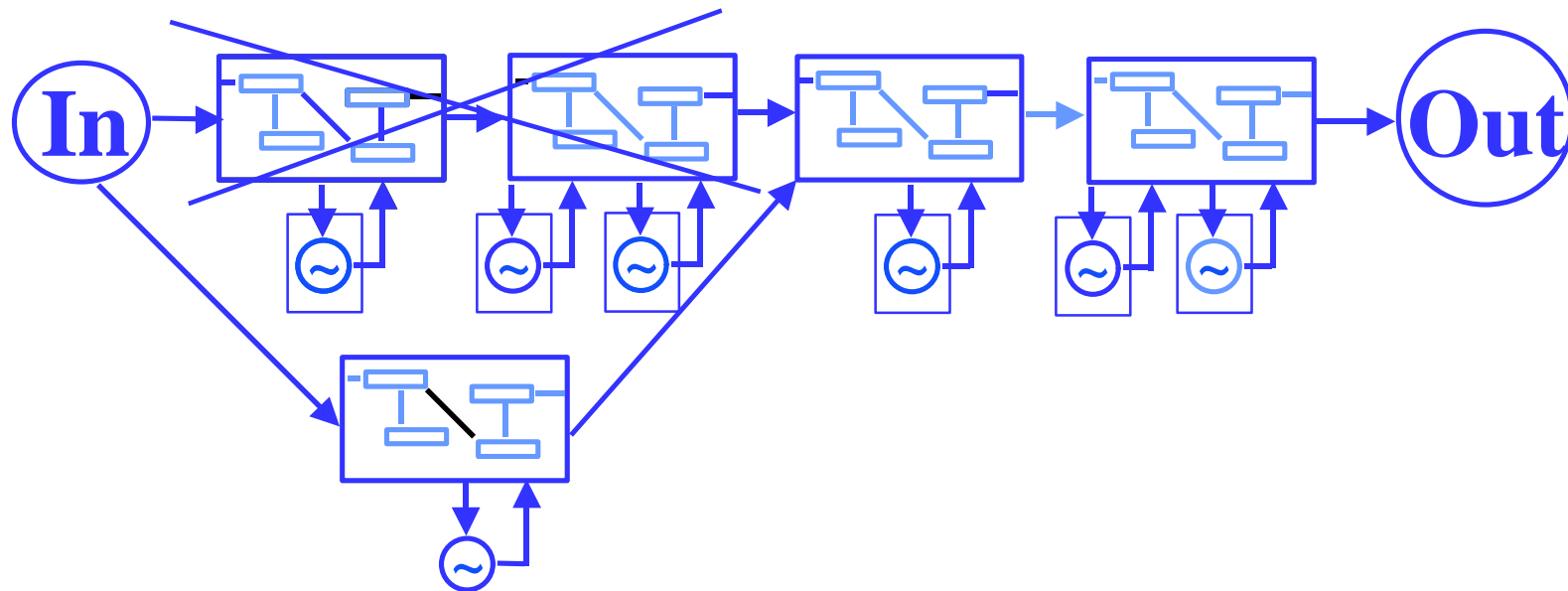
Automation and trying new technologies

Identify and eliminate chronic causes of poor performance

Continually improve the software process

The Optimizing Maturity Level

The Management View of the Software Process at Level 5



The software process is continuously improved in a controlled manner

The Optimizing Maturity Level

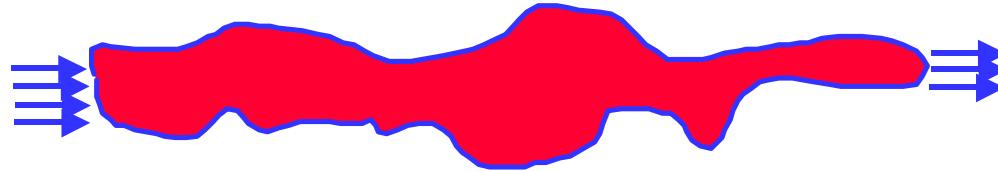
The Key Process Areas for the Optimizing Level (5)

Optimizing (5)

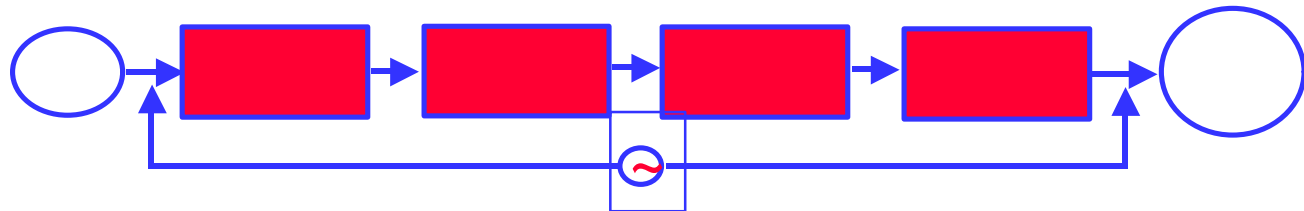
**Process change management
Technology change management
Defect prevention**

Management View of the Maturity Levels

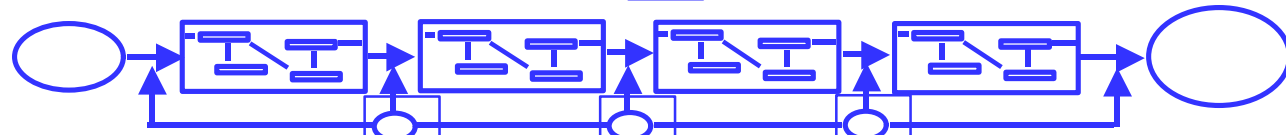
ML 1



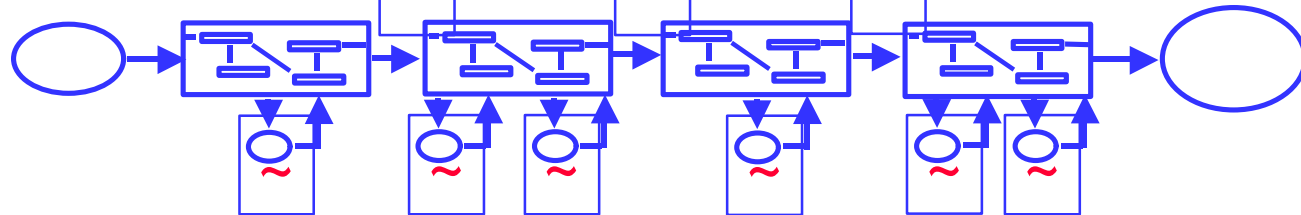
ML 2



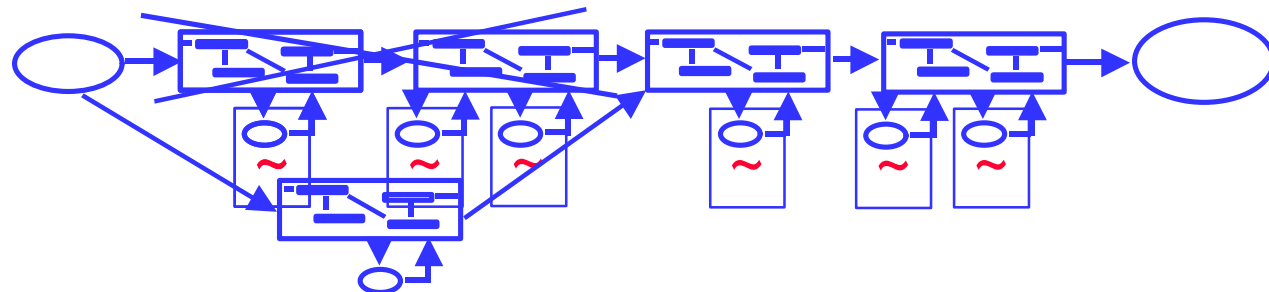
ML 3



ML 4

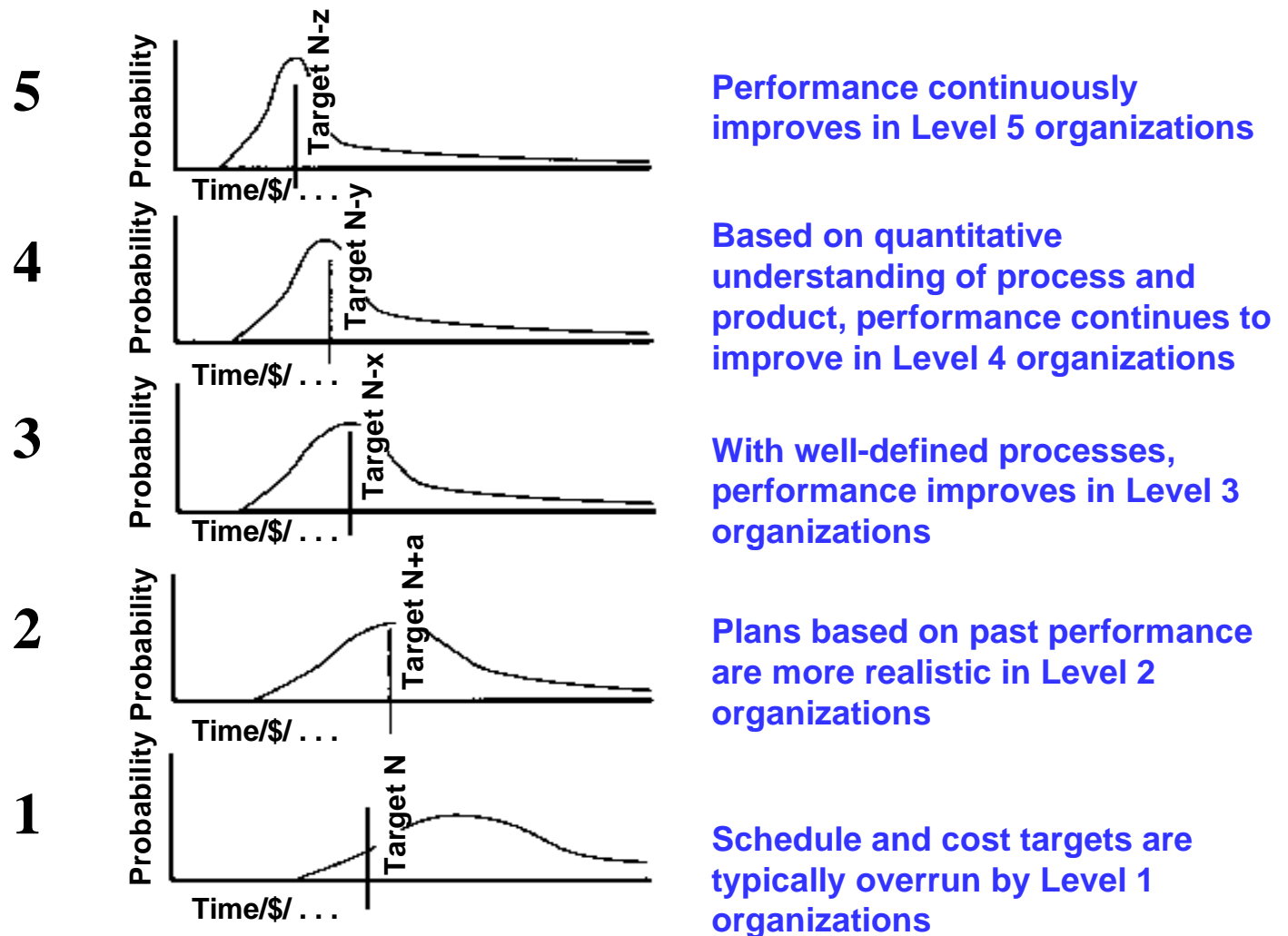


ML 5



Process Capability as Indicated by Maturity Level

Levels



A Foundation, Not a Destination

- The optimizing level is not the destination of process management.
- The destination is better products for a better price.
- The optimizing level is a foundation for building ever-improving capability.

from SEI Training

Process Improvement is Continuous Improvement

- We can never reach perfection.
- The CMM does not provide all the answers; it too is evolving and improving.
- Process management means constructive and continual improvement.
- The focus is on always doing better.
- Our reach should always exceed our grasp

from SEI Training

Quality is NOT Free...

Improvement requires investment.

- a vision of what you want
- commitment, planning and resources

... but quality is cheaper than the alternatives

The Capability Maturity Model provides a framework for continuous process improvement.

from SEI Training